

1/5/1 (Item 1 from file: 351)  
DIALOG(R)File 351:Derwent WPI  
(c) 2003 Thomson Derwent. All rts. reserv.

012874383 \*\*Image available\*\*  
WPI Acc No: 2000-046216/ 200004  
XRPX Acc No: N00-035791

Data decoding method in electronic commercial transaction system -  
involves decoding data attached with key recovery demand, using searched  
disclosure key of one of disclosure key certificate

Patent Assignee: HITACHI LTD (HITA )

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 11308213	A	19991105	JP 98109450	A	19980420	200004 B

Priority Applications (No Type Date): JP 98109450 A 19980420

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
JP 11308213	A	19	H04L-009/32	

Abstract (Basic): JP 11308213 A

NOVELTY - The user is authenticated, using specific disclosure key certificate received along with key recovery demand. Then, the disclosure key of another disclosure key certificate stored in memory, is searched using two secret keys. The data attached with the recovery demand are decoded, using searched disclosure key. DETAILED DESCRIPTION - A specific public-presentation key certificate is canceled, according to the demand of user. Then, a new public presentation key certificate is published, when the demand is made from the user. The key recovery demand including the specific disclosure key certificate and the data encoded using disclosure key of another disclosure key certificate. An INDEPENDENT CLAIM is also included for data decoding apparatus used in electronic commercial transaction system.

USE - For decoding data in electronic commercial transaction system.

ADVANTAGE - Since management of disclosure key certificate and key decoding are performed using single unit, a pair of secret keys are made available for only the user to recognize the encoded data when disclosure key of disclosure key certificate is lost. DESCRIPTION OF

DRAWING(S) - The figure shows schematic block diagram of a system for which the data decoding method is adopted.

Dwg.1/10

Title Terms: DATA; DECODE; METHOD; ELECTRONIC; COMMERCIAL;  
TRANSACTION;

SYSTEM; DECODE; DATA; ATTACH; KEY; RECOVER; DEMAND; SEARCH;  
DISCLOSE; KEY

; ONE; DISCLOSE; KEY; CERTIFY

Derwent Class: P85; W01

International Patent Class (Main): H04L-009/32

International Patent Class (Additional): G09C-001/00; H04L-009/08

File Segment: EPI; EngPI

1/5/2 (Item 1 from file: 347)  
DIALOG(R)File 347:JAPIO  
(c) 2003 JPO & JAPIO. All rts. reserv.

06366602 \*\*Image available\*\*  
ENCRYPTION DATA RECOVERY METHOD AND ITS SYSTEM

PUB. NO.: 11-308213 A]  
PUBLISHED: November 05, 1999 (19991105)  
INVENTOR(s): DOMYO SEIICHI  
UMEKI HISASHI  
TSUCHIYA HIROYOSHI  
KAWAI TORU  
YANAGIUCHI HIDETAKA  
APPLICANT(s): HITACHI LTD  
APPL. NO.: 10-109450 [JP 98109450]  
FILED: April 20, 1998 (19980420)  
INTL CLASS: H04L-009/32; G09C-001/00; H04L-009/08

#### ABSTRACT

**PROBLEM TO BE SOLVED:** To allow the system to approve data recovery by using a secret key only when a user receiving an issued public key certificate loses the secret key in pairs with an open key in the certificate by unifying the user management of a public key certificate and key recovery.

**SOLUTION:** A certification authority CA 220 abolishes a 1st public key certificate on request of a user device USC 200 and issues a 2nd public key certificate. Furthermore, a data recovery device DRC 240 authenticates the user by using the 2nd public key certificate attached to a key recovery request according to the key recovery request of the USC 200 and uses a secret key in pairs with the public key of the 1st public key certificate deposited by a key deposit device KEC 230 to decode the data encrypted by the open key of the 1st public key certificate.

COPYRIGHT: (C)1999,JPO

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-308213

(43) 公開日 平成11年(1999)11月5日

(51) Int.Cl.<sup>9</sup>

識別記号

F I

H 0 4 L 9/32

H 0 4 L 9/00

6 7 5 D

G 0 9 C 1/00

6 3 0

G 0 9 C 1/00

6 3 0 F

H 0 4 L 9/08

H 0 4 L 9/00

6 0 1 F

審査請求 未請求 請求項の数10 O L (全 19 頁)

(21) 出願番号

特願平10-109450

(22) 出願日

平成10年(1998)4月20日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 道明 誠一

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

(72) 発明者 梅木 久志

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

(72) 発明者 土屋 宏嘉

神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内

(74) 代理人 弁理士 富田 和子

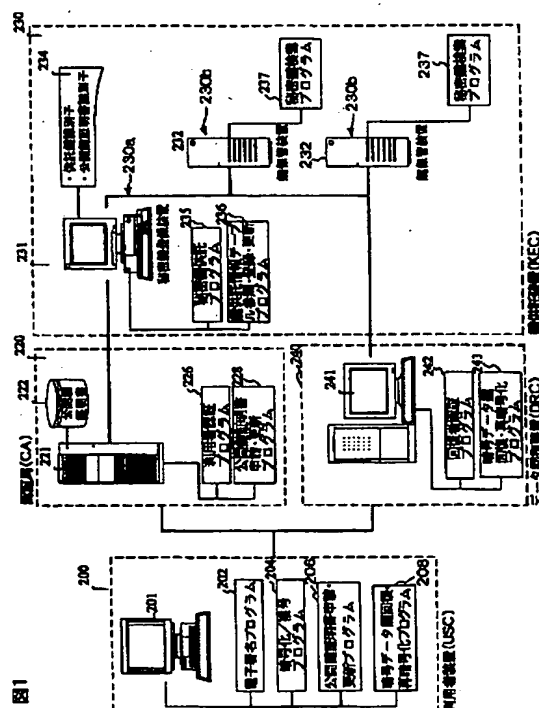
最終頁に続く

(54) 【発明の名称】 暗号データ回復方法および装置

(57) 【要約】

【課題】 公開鍵証明書および鍵回復の利用者管理を一元化し、公開鍵証明書の発行を受けた利用者が当該証明書の公開鍵と対の秘密鍵を紛失した場合にのみ、当該秘密鍵を用いたデータ回復を認めるようにする。

【解決手段】 CA 220 は、USC 200 の要求にしたがい第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する。また、DRC 240 は、USC 200 の鍵回復要求にしたがい、当該鍵回復要求に付された第2の公開鍵証明書を用いて認証した後、KEC 230 に供託された第1の公開鍵証明書の公開鍵と対の秘密鍵を用いて、前記第1の公開鍵証明書の公開鍵で暗号化されたデータを復号する。



## 【特許請求の範囲】

【請求項1】 認証局が発行した公開鍵証明書と対の秘密鍵により暗号化されたデータを回復する暗号データ回復方法であって、

認証局において、

利用者の要求にしたがい、第1の公開鍵証明書を発行する第1のステップと、

前記利用者の要求にしたがい、当該第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する第2のステップと、

前記利用者からの、前記第1の公開鍵証明書の公開鍵により暗号化されたデータおよび前記第2の公開鍵証明書を用いた鍵回復要求を受け付ける第3のステップと、前記第3のステップで受け付けた鍵回復要求にしたがい、当該鍵回復要求に付された前記第2の公開鍵証明書を用いて、前記利用者を認証する第4のステップと、前記第4のステップにより、前記利用者が認証された場合に、前記第1の公開鍵証明書の公開鍵と対の秘密鍵を、供託されている鍵の中から検索する第5のステップと、

前記第5のステップで検索した秘密鍵を用いて、前記鍵回復要求に付された前記暗号化されたデータを回復する第6のステップと、を備えることを特徴とする暗号データ回復方法。

【請求項2】 請求項1記載の暗号データ回復方法であって、

前記第5のステップは、前記第1の公開鍵証明書が前記利用者に複数発行されている場合、前記複数の第1の公開鍵証明書各々の有効期限と、前記鍵回復要求に付された前記暗号化されたデータの暗号化時刻とを比較して、当該暗号化されたデータを暗号化するのに用いた秘密鍵を、供託されている鍵の中から検索することを特徴とする暗号データ回復方法。

【請求項3】 認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵により暗号化されたデータを、当該秘密鍵を供託された鍵回復センタにより回復する暗号データ回復方法であって、

認証局において、

利用者の要求にしたがい、第1の公開鍵証明書を発行する第1のステップと、

前記利用者の要求にしたがい、当該第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する第2のステップと、

前記第2のステップで廃止した前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子を、当該利用者に送付する第3のステップと、

鍵回復センタにおいて、

前記利用者からの、前記第1の公開鍵証明書の公開鍵により暗号化されたデータと前記供託鍵識別子とが連結されたデータに、前記第2の公開鍵証明書の公開鍵と対の

秘密鍵による署名が付されたデータを、鍵回復要求として受け付ける第4のステップと、

前記第4のステップで受け付けた鍵回復要求に付された前記署名を、前記第2の公開鍵証明書の公開鍵を用いて検証することで、前記利用者を認証する第5のステップと、

前記第5のステップにより、前記利用者が認証された場合に、前記第4のステップで受け付けた鍵回復要求に付された供託鍵識別子が示す秘密鍵を、供託されている鍵の中から検索する第6のステップと、

前記第6のステップで検索した秘密鍵を用いて、前記鍵回復要求に付された、前記暗号化されたデータを回復する第7のステップと、を備えることを特徴とする暗号データ回復方法。

【請求項4】 請求項3記載の暗号データ回復方法であって、

鍵回復センタにおいて、

前記第7のステップは、回復したデータを、前記第2の公開鍵証明書の公開鍵を用いて再暗号化して利用者に提供することを特徴とする暗号データ回復方法。

【請求項5】 請求項3記載の暗号データ回復方法であって、

認証局において、

前記第2のステップは、廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を用いた鍵回復の有効期限を設定し、

鍵回復センタにおいて、

前記第5のステップは、前記第4のステップで受け付けた鍵回復要求が、前記第2のステップで設定した有効期限を過ぎている場合、前記認証処理を行わないことを特徴とする暗号データ回復方法。

【請求項6】 請求項3記載の暗号データ回復方法であって、

認証局において、

前記第3のステップは、前記第2のステップで廃止した前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子を、当該供託鍵識別子に有効期限を設定して利用者に送付し、

鍵回復センタにおいて、

前記第5のステップは、前記第4のステップで受け付けた鍵回復要求に付された前記暗号化されたデータの暗号化時刻が、当該鍵回復要求に付された供託鍵識別子の有効期限を過ぎている場合、前記認証処理を行わないことを特徴とする暗号データ回復方法。

【請求項7】 認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵により暗号化されたデータを回復する暗号データ回復システムであって、

鍵供託装置と、利用者装置と、データ回復装置とを備え、

前記鍵供託装置は、

10

20

30

40

50

## 3

認証局が廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を保管する鍵保管手段と、  
 前記データ回復装置の指示にしたがい、当該装置から通知された供託鍵識別子が示す秘密鍵を前記鍵保管手段から検索し、前記データ回復装置に通知する通知手段と、  
 を有し、  
 前記利用者装置は、  
 前記第1の公開鍵証明書の公開鍵で暗号化されたデータと、認証局から送付された前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子とを連結し、前記連結したデータに対して、認証局が発行した前記第2の公開鍵証明書の公開鍵と対の秘密鍵により電子署名を生成し、当該電子署名がなされた前記連結したデータを、前記データ回復装置に通知する電子署名生成手段を有し、  
 前記データ回復装置は、  
 前記利用者装置から通知された前記電子署名を、前記第2の公開鍵証明書の公開鍵を用いて検証することで、利用者を認証する認証手段と、  
 前記認証手段により、利用者が認証された場合に、前記利用者装置から通知された前記連結したデータを構成する供託鍵識別子を、前記鍵供託装置に通知して、当該供託鍵識別子が示す秘密鍵を取得する取得手段と、  
 前記取得手段により取得した秘密鍵を用いて、前記利用者装置から通知された前記連結したデータを構成する暗号化されたデータを回復する回復手段と、を有することを特徴とする暗号データ回復システム。  
 【請求項8】 認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵により暗号化されたデータを回復する暗号データ回復装置であって、  
 認証局が廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を保管する鍵保管手段と、  
 前記第2の公開鍵証明書の公開鍵と対の秘密鍵により電子署名がなされた、前記第1の公開鍵証明書の公開鍵で暗号化されたデータと前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子とが連結されたデータを、利用者から受け取る受信手段と、  
 前記第2の公開鍵証明書の公開鍵を用いて、前記連結されたデータに付された電子署名を検証することで、利用者を認証する認証手段と、  
 前記認証手段により、利用者が認証された場合に、前記受信手段により受信した前記連結されたデータを構成する供託鍵識別子が示す秘密鍵を、前記鍵保管手段から取得する取得手段と、  
 前記取得手段により取得した秘密鍵を用いて、前記受信手段により受信した前記連結されたデータを構成する暗号化されたデータを回復する回復手段と、を備えていることを特徴とする暗号データ回復装置。  
 【請求項9】 認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵により暗号化されたデータを回復させるため

## 4

のプログラムが記憶された記憶媒体であって、  
 当該プログラムは、情報処理装置に、  
 認証局が廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を記憶装置に保管する第1のステップと、  
 前記第2の公開鍵証明書の公開鍵と対の秘密鍵により電子署名がなされた、前記第1の公開鍵証明書の公開鍵で暗号化されたデータと前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子とが連結されたデータを、利用者から受け取る第2のステップと、  
 前記第2の公開鍵証明書の公開鍵を用いて、前記第2のステップにより受信した前記連結されたデータに付された電子署名を検証することで、利用者を認証する第3のステップと、  
 前記第3のステップにより、利用者が認証された場合に、前記第2のステップにより受信した前記連結されたデータを構成する供託鍵識別子が示す秘密鍵を、記憶装置から取得する第4のステップと、  
 前記第4のステップにより取得した秘密鍵を用いて、前記第2のステップにより受信した前記連結されたデータを構成する暗号化されたデータを回復する第5のステップと、を実行させることを特徴とするプログラムが記憶された記憶媒体。  
 【請求項10】 送信者の装置において、  
 取引データを、認証局が発行した第1の公開鍵証明書の公開鍵を用いて暗号化し、受信者に送信するステップと、  
 受信者の装置において、  
 前記送信者から送信された前記暗号化された取引データを記憶媒体に保管するステップと、  
 前記記憶媒体に保管した前記暗号化された取引データを、前記第1公開鍵証明書の公開鍵と対の秘密鍵を用いて復号するステップと、を備える電子商取引方法であって、  
 認証局において、  
 前記受信者の要求にしたがい、当該第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する第1のステップと、  
 前記受信者からの、前記暗号化された取引データおよび前記第2の公開鍵証明書を用了鍵回復要求を受け付ける第2のステップと、  
 前記第2のステップで受け付けた鍵回復要求にしたがい、当該鍵回復要求に付された前記第2の公開鍵証明書をを用いて、前記受信者を認証する第3のステップと、  
 前記第3のステップにより、前記受信者が認証された場合に、前記第1の公開鍵証明書の公開鍵と対の秘密鍵を、供託されている鍵の中から検索する第4のステップと、  
 前記第4のステップで検索した秘密鍵を用いて、前記鍵回復要求に付された前記暗号化された取引データを回復する第5のステップと、を備えることを特徴とする電子

商取引方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、利用者の公開鍵で暗号化されたデータを、当該利用者の要求に応じて復号化する暗号データ回復技術、特に、エンベロープ (Envelope) 型暗号システムにおいて、利用者の公開鍵でカプセル化されたセッション鍵を、当該利用者の要求に応じてカプセル開放する鍵回復技術に関する。

【0002】

【従来の技術】暗号技術は、情報の盗聴や改ざん、あるいは利用者のなりすましなどを防止する計算機技術である。データを暗号化する効果の一つとして、情報にアクセス(復号)できるのは、復号用の鍵を持つ利用者に限定されることを挙げることができる。データの暗号化は、計算機システムの運用(オン/オフライン)やプラットフォーム(OS)に依存することなく、機密情報のアクセス制御を容易に実現する手段といえる。

【0003】機密情報の保管に関し、通常、以下の条件が課せられる。

【0004】(1) 法律や組織の規定によって長期間(たとえば年単位)保存の義務がある。

【0005】(2) いったん保存したら、ほとんどアクセスすることがない。

【0006】(3) 外部あるいは社内の監査時には、確実に提出(復号)できるものでなければならぬ。

【0007】したがって、機密情報を提出しようとしたが復号できない場合は、組織や担当部署の不利益となることが予想される。復号できない理由としては、保存したはずの機密情報が見つからない場合の他に、復号用の鍵を紛失した場合や、担当者の変更により鍵の所在がわからなくなった場合など、鍵管理に問題がある場合が考えられる。

【0008】従来、暗号システムの鍵管理は、利用者やシステム運用者(とくに機密情報の保存責任者)による自主的な管理に任せている場合が多い。したがって、機密情報の暗号化がさらに普及するにつれ、鍵管理をめぐるトラブルが多発することが予想される。

【0009】そこで、復号用の鍵を、利用者ではなくシステムで一元管理し、いざというときに取出すバックアップシステムが提案されている。この技術は、鍵回復技術と呼ばれている。とくに、鍵回復技術のうち、利用者の秘密鍵をシステムに供託するタイプの方式は、鍵供託方式と呼ばれている。

【0010】鍵供託方式では、利用者端末、鍵供託装置(鍵保管装置)、およびデータ回復装置など、機能ごとに専門の計算機装置を配置する分散システムを構築している。このような鍵回復方式の公知例として、たとえば「暗号データ回復方法、鍵登録システムおよびデータ回復システム(特願平9-80081号)」がある。

【0011】また、公開鍵暗号アルゴリズムを用いたシステムでは、秘密鍵の安全管理、および公開鍵の信頼性確保を図るため、公開鍵証明書を発行する専門機関である認証局を設立し、当該認証局が発行した公開鍵証明書を用いて認証を行う利用者認証技術が提案されている。当該技術では、利用者は、認証局ならびに公開鍵証明書を信頼することで、相手を認証したり、メッセージの完全性を確認したりすることができる。このような利用者認証方式の公知例として、たとえば「暗号通信における公開鍵登録方法および公開鍵証明書の発行局(特開平7-160198)」がある。

【0012】

【発明が解決しようとする課題】ところで、上記説明した従来の鍵回復方式および利用者認証方式には、以下のような問題がある。

【0013】従来は、鍵の供託および供託された鍵を用いてデータの回復を行う機関(鍵回復システム)と、公開鍵証明書を発行する認証局とが別個独立して設けられている。このため、公開鍵証明書により証明された公開鍵と対の秘密鍵を供託しようとする場合、利用者は、まず認証局に公開鍵証明書を発行してもらい、次に、当該公開鍵証明書の公開鍵と対の秘密鍵を鍵回復システムに供託する。つまり、利用者は、認証局および鍵回復システムに対して、別個独立に手続きを行うことになる。

【0014】したがって、認証局および鍵回復システムは、各々の利用者を互いに無関係に管理することになる(2元管理)。このため、鍵回復システムに供託した、認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵を紛失(ここでいう紛失とは、秘密鍵が格納されたICカードを物理的に紛失した場合の他に、秘密鍵を格納した鍵ファイルをオープンするパスワードを忘れる場合など、事実上、秘密鍵が使用不可能な状態も含む)した場合、利用者は、鍵回復センタに対して、紛失した秘密鍵を用いた暗号データの復号を依頼することになる。この場合、認証局からすれば、利用者の管理を鍵回復システムとは別個独立に行っているため、利用者からその事実(認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵を紛失したこと)が伝えられない限り、把握することができない。

【0015】また、紛失した秘密鍵を用いた暗号データの復号を依頼する利用者が、当該秘密鍵と対になる公開鍵の公開鍵証明書の発行を受けた者であるという保証もない。つまり、認証局および鍵回復センタ各々に登録された利用者が同一人物であるという保証がない。

【0016】本発明は、上記事情に鑑みてなされたものであり、認証局および鍵回復システムにおける利用者管理を一元化し、認証局による公開鍵証明書の発行を受けた利用者に対してのみ、当該公開鍵証明書の公開鍵と対の秘密鍵を紛失した場合に、当該秘密鍵を用いたデータ回復を認めるデータ回復方法を提供することにある。

【0017】

【課題を解決するための手段】上記課題を解決するために、本発明の第1の態様は、認証局が発行した公開鍵証明書  
の公開鍵と対の秘密鍵により暗号化されたデータを回復する暗号データ回復方法であって、認証局において、利用者の要求にしたがい、第1の公開鍵証明書を発行する第1のステップと、前記利用者の要求にしたがい、当該第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する第2のステップと、前記利用者からの、前記第1の公開鍵証明書の公開鍵により暗号化されたデータおよび前記第2の公開鍵証明書をを用いた鍵回復要求を受け付ける第3のステップと、前記第3のステップで受け付けた鍵回復要求にしたがい、当該鍵回復要求に付された前記第2の公開鍵証明書をを用いて、前記利用者を認証する第4のステップと、前記第4のステップにより、前記利用者が認証された場合に、前記第1の公開鍵証明書の公開鍵と対の秘密鍵を、供託されている鍵の中から検索する第5のステップと、前記第5のステップで検索した秘密鍵を用いて、前記鍵回復要求に付された前記暗号化されたデータを回復する第6のステップと、を備えることを特徴とする。

【0018】上記第1の態様によれば、認証局は、第1の公開鍵証明書（データ暗復号用）を発行した利用者が当該証明書の公開鍵と対の秘密鍵を紛失した場合、当該利用者から要求があった場合にのみ、当該証明書を廃止して、前記利用者に第2の公開鍵証明書（認証用）を発行するようにしている。

【0019】そして、前記第2の公開鍵証明書をを用いて利用者を認証することにより、前記第2の公開鍵証明書を発行を受けた利用者にのみ、第1の公開鍵証明書の公開鍵と対の秘密鍵を用いたデータ復号（たとえば、セッション鍵のカプセル開放）を認めるようにしている。

【0020】このようにすることで、鍵回復を公開鍵証明書の利用者にのみ認めることになり、このため、公開鍵証明書の利用者の認証・登録手段と鍵回復の利用者の認証・登録手段とを共通化することができるので、利用者管理を一元化することができる。また、認証局による公開鍵証明書の発行を受けた利用者に対してのみ、当該公開鍵証明書の公開鍵と対の秘密鍵を紛失した場合に、当該秘密鍵を用いたデータ回復を認めることができる。

【0021】なお、上記第1の態様において、前記第5のステップは、前記第1の公開鍵証明書が前記利用者に複数発行されている場合、前記複数の第1の公開鍵証明書各々の有効期限と、前記鍵回復要求に付された前記暗号化されたデータの暗号化時刻とを比較して、当該暗号化されたデータを暗号化するのに用いた秘密鍵を検索するものでもよい。

【0022】このようにすることで、たとえば、利用者が、認証局からデータ暗復号用の第1の公開鍵証明書を複数発行してもらっている場合において、相手から、そ

のうちの1つの公開鍵を用いて暗号化されたデータを受け取った場合、対応する秘密鍵が分からなくなってしまうデータを復号できない場合でも、認証局に鍵回復要求をすることで、当該データの回復を行うことができる。

【0023】また、本発明の第2の態様は、認証局が発行した公開鍵証明書の公開鍵と対の秘密鍵により暗号化されたデータを、当該秘密鍵を供託された鍵回復センタにより回復する暗号データ回復方法であって、認証局において、利用者の要求にしたがい、第1の公開鍵証明書を発行する第1のステップと、前記利用者の要求にしたがい、当該第1の公開鍵証明書を廃止して、第2の公開鍵証明書を発行する第2のステップと、前記第2のステップで廃止した前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子を、当該利用者に送付する第3のステップと、鍵回復センタにおいて、前記利用者からの、前記第1の公開鍵証明書の公開鍵により暗号化されたデータと前記供託鍵識別子とが連結されたデータに、前記第2の公開鍵証明書の公開鍵と対の秘密鍵による署名が付されたデータを、鍵回復要求として受け付ける第4のステップと、前記第4のステップで受け付けた鍵回復要求に付された前記署名を、前記第2の公開鍵証明書の公開鍵を用いて検証することで、前記利用者を認証する第5のステップと、前記第5のステップにより、前記利用者が認証された場合に、前記第4のステップで受け付けた鍵回復要求に付された供託鍵識別子が示す秘密鍵を、供託されている鍵の中から検索する第6のステップと、前記第6のステップで検索した秘密鍵を用いて、前記鍵回復要求に付された、前記暗号化されたデータを回復する第7のステップと、を備えることを特徴とする。

【0024】本発明の第2の態様においても、第1の態様と同様、公開鍵証明書の利用者の認証・登録手段と鍵回復の利用者の認証・登録手段とを共通化することができるので、利用者管理を一元化することができる。また、認証局による公開鍵証明書の発行を受けた利用者に対してのみ、当該公開鍵証明書の公開鍵と対の秘密鍵を紛失した場合に、当該秘密鍵を用いたデータ回復を認めることができる。

【0025】なお、上記第2の態様において、前記第2の公開鍵証明書を利用者認証用兼データ暗号化用として用い、前記第7のステップにおいて、回復したデータを、前記第2の公開鍵証明書の公開鍵を用いて再暗号化して利用者に提供するようにしてもよい。

【0026】このようにすることで、回復したデータを利用者に通知する際のセキュリティを向上させることができる。

【0027】また、前記第2のステップにおいて、廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を用いた鍵回復の有効期限を設定するようにし、前記第5のステップにおいて、前記第4のステップで受け付けた鍵回復

要求が前記2のステップで設定した有効期限を過ぎている場合、前記データ回復処理を行わないようにしてもよい。

【0028】あるいは、前記第3のステップにおいて、前記第2のステップで廃止した前記第1の公開鍵証明書の公開鍵と対の秘密鍵を示す供託鍵識別子を、当該供託鍵識別子に有効期限を設定して利用者に送付するようにし、前記第5のステップにおいて、前記第4のステップで受け付けた鍵回復要求に付された前記暗号化されたデータの暗号化時刻が当該鍵回復要求に付された供託鍵識別子の有効期限を過ぎている場合、前記データ回復処理を行わないようにしてもよい。

【0029】このようにすることで、鍵回復およびデータ回復を、第2の公開鍵証明書の発行（つまり、第1の公開鍵証明書の廃止）から所定期間にもみ認めるように限定することができる。

【0030】

【発明の実施の形態】以下に、本発明の第1実施形態について説明する。

【0031】図1は、本発明の第1実施形態が適用されたシステムの概略構成図である。

【0032】本実施形態が適用されたシステムは、図1に示すように、利用者装置（USC：User Security Component）200と、USC200からの要求に応じて公開鍵証明書を発行する認証局（CA：Certificate Authority）220と、CA220が発行した公開鍵証明書の公開鍵と対の秘密鍵を供託する鍵供託装置（KEC：Key Escrow Component）230と、USC200からの鍵回復要求に応じてKEC230に供託された秘密鍵を用いて、カプセル化されたセッション鍵を開放するデータ回復装置（DRC：Data Recovery Component）240と、を備えて構成されている。

【0033】本実施形態では、KEC230がCA220のバックエンド装置として設けられている。この点、従来の認証局および鍵回復システムと異なる。なお、図1では、CA220、KEC230、およびDRC240を、別個独立した装置として、分離して表示しているが、これは、従来の認証局および鍵回復システムの構成との差異を明確にするために、便宜上、そのように表示しているだけである。CA220、KEC230、およびDRC240を、1つの装置上で実現することにより、鍵回復機能を有する認証局としても構わない。

【0034】USC200は、計算機装置201において、電子署名プログラム202、暗号化／復号プログラム204、公開鍵証明書申請・更新プログラム206、および暗号データ鍵回復・再暗号化208がメモリ上にロードされ、CPUにより実行されることで実現される。

【0035】これらのプログラムは、たとえばCD-ROMなどの記憶媒体により、計算機装置201に提供さ

れるものでもよい。あるいは、ICカードなどの計算機能を備えた記憶媒体に格納しておき、計算機装置201に当該記憶媒体が装着されると、当該記憶媒体自体がこれらのプログラムを実行するようにしてもよい。

【0036】CA220は、公開鍵証明書を格納するデータベース（DB）222を備えた計算機装置221において、利用者認証プログラム226、および公開鍵証明書申請・更新プログラム228がメモリ上にロードされ、CPUにより実行されることで実現される。

【0037】KEC230は、供託の対象となる秘密鍵を登録する秘密鍵登録装置230aと、秘密鍵登録装置230aにより登録された秘密鍵を保管するとともに、秘密鍵登録装置230aからの指示にしたがい保管した秘密鍵の検索を行う鍵保管装置230bと、で構成される。

【0038】秘密鍵登録装置230aは、計算機装置231において、CA220から送られてきた秘密鍵を、鍵保管装置230bに登録（供託）するための秘密鍵供託プログラム235、および、秘密鍵を供託した証拠としての付加情報（鍵供託識別子）と当該秘密鍵と対の公開鍵についての公開鍵証明書を示す識別子とが互いに関連付けられたデータを記憶する鍵供託情報テーブル234の参照・登録・更新処理を行う鍵供託情報テーブル参照・登録・更新プログラム236が、メモリ上にロードされ、CPUにより実行されることで実現される。

【0039】鍵保管装置230bは、計算機装置232において、秘密鍵登録装置230aからの指示にしたがい、保管した秘密鍵から所定の秘密鍵を検索するための秘密鍵検索プログラム237がメモリ上にロードされ、CPUにより実行されることで実現される。

【0040】なお、秘密鍵登録装置230aおよび鍵保管装置230bは、1つの装置上で実現されるようにしてもよい。

【0041】DRC240は、計算機装置241において、USC200を用いて鍵回復要求を行った利用者の認証を行うための回復者認証プログラム242、および、USC200から送られてきた、カプセル化されたセッション鍵を開放し、当該鍵を再暗号化する暗号データ鍵回復・再暗号化プログラム243が、メモリ上にロードされ、CPUにより実行されることで実現される。

【0042】次に、以上説明したシステムの処理手順について説明する。

【0043】本実施形態の処理手順は、利用者が、CA220に第1の公開鍵証明書の発行を依頼してから、当該証明書の廃止して第2の公開鍵証明書を再発行してもらうまでの処理手順と、利用者が、再発行を受けた第2の公開鍵証明書を用いて認証を受けることで、廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵をカプセル開放してもらうまでの処理手順とに分けて考えることができる。



【0044】まず、利用者が、CA220に第1の公開鍵証明書の発行を依頼してから、当該証明書の廃止して第2の公開鍵証明書を再発行してもらうまでの処理手順について、図2を参照して説明する。

【0045】図2は、本発明の第1実施形態において、第1の公開鍵証明書を発行してから、当該証明書を廃止して第2の公開鍵証明書を再発行するまでの処理手順を説明するためのフロー図である。

【0046】まず、USC200は、利用者からの鍵生成指示にしたがい、暗号化／復号化プログラム204を  
10 実行して、公開鍵暗号における公開鍵と秘密鍵とを生成する（ステップ100）。たとえば、利用者が、計算機装置201が備えるキーボード、マウスなどの入力装置を用いて乱数を投入すると、計算機装置201は、その乱数を鍵のシードとして、公開鍵暗号の公開鍵と秘密鍵とを生成し、これを計算機装置201が備えるハードディスクなどの記憶装置に鍵ファイルとして格納する。

【0047】次に、USC200は、公開鍵証明書申請・更新プログラム206の実行を開始し、ステップ100で生成した公開鍵および秘密鍵を、CA220に送付  
20 する。これにより、第1の公開鍵証明書の発行を申請する（ステップ102）。なお、公開鍵および秘密鍵の送付方法は、CA220において、適当な利用者認証を利用することができるのであれば、オンラインおよびオフラインのいずれでもよい。

【0048】CA220は、USC200から第1の公開鍵証明書の発行申請を受け取ると、公開鍵証明書申請・更新プログラム226の実行を開始し、第1の公開鍵証明書を発行して、これをUSC200に返信する。また、発行した第1の公開鍵証明書をデータベース222  
30 に格納する（ステップ104）。

【0049】次に、USC200は、CA220が第1の公開鍵証明書を発行したことを確認すると、当該証明書の公開鍵と対の秘密鍵を供託するか否かについて、利用者に指示を催促する。そして、受け付けた指示をCA220に通知する（ステップ108）。その後、公開鍵証明書申請・更新プログラム206の実行を一旦終了する。

【0050】CA220は、USC200から受け取った通知が供託する旨を示している場合、ステップ102  
40 で受け取った秘密鍵と、ステップ104で発行した第1の公開鍵証明書（あるいは、当該証明書の識別子）とを、KEC230に送付する（ステップ110）。その後、公開鍵証明書申請・更新プログラム228の実行を一旦終了する。

【0051】KEC230は、秘密鍵および第1の公開鍵証明書（あるいは、当該証明書の識別子）を受け取ると、秘密鍵供託プログラム235を実行し、鍵保管装置230bのいずれか一方、あるいは両方に当該秘密鍵を  
50 保管する（ステップ110）。この際、セキュリティを

さらに向上させるために、Secret Sharingと呼ばれる暗号技術を用いて、秘密鍵を、当該秘密鍵を類推不可能なデータに分割して保管し、必要に応じて合成して、当該秘密鍵を復元するようにしてもよい。

【0052】なお、ステップ110での処理は、実際には、CA220のバックエンドに配置されたKEC230が行っているが、利用者からすれば、CA220が秘密鍵を保管しているように見える。

【0053】次に、KEC230は、鍵供託情報テーブル参照・登録・更新プログラム236を実行して、鍵を供託した証拠としての付加情報（鍵供託識別子）と第1の公開鍵証明書の識別子とを関連付けて、鍵供託情報テーブル234に格納する。

【0054】ここで、第三者あるいはUSC200の利用者が、第1の公開鍵証明書の公開鍵を用いてデータ暗号を行った場合（ステップ120）、具体的には、セッション鍵で暗号化したデータ（ステップ122）に、第1の公開鍵証明書の公開鍵を用いてカプセル化したセッション鍵（ステップ124）を添付した場合を想定する。そして、USC200の利用者がこのデータを復号  
する場合を考える。

【0055】第1の公開鍵証明書の公開鍵と対の秘密鍵を紛失していない場合（ステップ112）、USC200は、利用者からの復号指示にしたがい、暗号化／復号プログラム204を実行して、暗号データを復号化する。具体的には、第1の公開鍵証明書の公開鍵と対の秘密鍵を用いて、第1の公開鍵証明書の公開鍵によりカプセル化されたセッション鍵をカプセル開放し、それから、カプセル開放されたセッション鍵を用いて、暗号データ  
を復号化する。

【0056】一方、第1の公開鍵証明書の公開鍵と対の秘密鍵を紛失してしまった場合（ステップ112）、USC200は、第1の公開鍵証明書の公開鍵によりカプセル化されたセッション鍵をカプセル開放することができない。

【0057】この場合、ステップ130に示す公開鍵証明書再発行処理が実行されることになる。

【0058】まず、USC200は、利用者からの公開鍵証明書再発行指示にしたがい、公開鍵証明書申請・更新プログラム206を再度実行して、CA220に対し、ステップ104にて発行された第1の公開鍵証明書の無効化（廃止）を申請する（ステップ131）。

【0059】CA220は、第1の公開鍵証明書の無効申請を受け取ると、公開鍵証明書申請・更新プログラム228を再度実行して、データベース222に格納された当該第1の公開鍵証明書の内容を無効状態に更新し、当該証明書の廃止をUSC200に返信する。さらに、CA220は、関連部署に、CRL（Certificate Revocation List）とよばれる通知を行う（ステップ132）。

【0060】次に、USC200は、利用者からの鍵生成指示にしたがい、暗号化／復号化プログラム204を実行して、ステップ100と同様、公開鍵暗号における公開鍵と秘密鍵とを生成する（ステップ133）それから、公開鍵証明書申請・更新プログラム206により、ステップ102と同様、ステップ133で生成した公開鍵および秘密鍵を、CA220に送付する。これにより、公開鍵証明書の再発行（第2の公開鍵証明書の発行）を申請する（ステップ134）。なお、この再発行申請にあたり、ステップ132で行った第1の公開鍵証明書の廃止通知、あるいは廃止された第1の公開鍵証明書そのものを添付する必要があるかどうかは、システム運用によって異なることはいうまでもない。本実施形態では、添付しない場合について説明している。

【0061】次に、CA220は、USC200から公開鍵証明書の再発行申請を受け取ると、公開鍵証明書申請・更新プログラム226により、第2の公開鍵証明書を発行して、これをUSC200に返信する。また、発行した第2の公開鍵証明書をデータベース222に格納する（ステップ135）。この第2の公開鍵証明書は、ステップ133で生成した公開鍵についての証明書である。また、ステップ135において、CA220は、第2の公開鍵証明書と結合する方法および別途送付する方法のいずれかにより、第1の公開鍵証明書の公開鍵と対の秘密鍵がKEC230に供託された証拠を示す付加情報および当該秘密鍵を示す鍵供託識別子を、USC200に送付する。その後、フローを終了する。

【0062】なお、図2に示すフローでは、ステップ102において、公開鍵および秘密鍵を一括してCA220に提出し、第1の公開鍵証明書の発行を確認した後、ステップ108において、秘密鍵の供託をするか否かを判断するようにしている。しかしながら、ステップ108は、ステップ102の段階、すなわち、第1の公開鍵証明書の発行を申請する段階で行うようにしてもよい。あるいは、ステップ102では公開鍵のみを提出し、ステップ108において、USC200が電子署名プログラム202を実行して、第1の公開鍵証明書の公開鍵で検証可能な署名を生成して、当該署名と共に秘密鍵を供託するようにしてもよい。

【0063】次に、上記説明したステップ130（公開鍵証明書更新処理）におけるデータの流れについて説明する。

【0064】なお、以下の図の説明では、鍵Kを用いてデータXを暗号化することをE[K](X)と記す。

【0065】図3は、図2のステップ130（公開鍵証明書の更新手順）におけるデータの流れを概念的に示した図である。

【0066】図3において、USC200は、CA220に対して、第1の公開鍵証明書の無効化要求301を送信する。当該要求301には、一般にリボケーション

要求と呼ばれる申請CertRevoke-reqを乱数rで暗号化したものと、乱数rをCA220の公開鍵CApubでカプセル化したものが含まれる。

【0067】CA220は、第1の公開鍵証明書の無効化要求301を受信すると、CA220の秘密鍵CApriを用いて乱数rを取り出し、さらに申請CertRevoke-reqを復号する。そして、申請CertRevoke-reqの内容を審理し、OKならば、USC200に、無効化OKの応答302を返す。当該応答302には、申請結果通知CertRevoke-resが含まれる。このとき、USC200には、信頼できる公開鍵がまだ存在してない。そこで、ワンタイムな認証手続きの実現手段として、たとえば、USC200から送られてきた乱数rを基に、申請結果通知CertRevoke-resを暗号化することが好ましい。

【0068】次に、USC200は、CA220に、公開鍵証明書の再発行（第2の公開鍵証明書の発行）要求311を送信する。当該要求311には、一般にリニューアル要求と呼ばれる申請CertRenewal-reqを乱数r2で暗号化したものと、新しく生成した公開鍵UserAnewPub（オプションでは秘密鍵UserAnewpri）を乱数r2で暗号化したものと、乱数r2をCA220の公開鍵CApubでカプセル化したものが含まれる。

【0069】CA220は、公開鍵証明書の再発行（第2の公開鍵証明書の発行）要求311を受信すると、CA220の秘密鍵CApriを用いて乱数r2を取り出し、さらに申請CertRenewal-reqを復号する。ここで、第1の公開鍵証明書の無効化要求301と公開鍵証明書の再発行要求311とが、同じセッションで行われるのであれば、乱数r2に乱数rと同じものを用いてもよい。

【0070】次に、CA220は、復号した申請CertRenewal-reqの内容を審理し、OKならば、KEC230に、無効化された第1の公開鍵証明書の公開鍵と対の秘密鍵についての鍵回復有効要求321を送信する。当該要求321には、無効化された第1の公開鍵証明書の公開鍵UserAoldPubおよび当該公開鍵と対の秘密鍵を用いた復号を可能にするための申請KECActive-reqを、乱数r3で暗号化したものと、乱数r3をKEC230の公開鍵KECpubでカプセル化したものが含まれる。

【0071】KEC230は、鍵回復有効要求321を受信すると、KEC230の秘密鍵KECpriを用いて乱数r3を取り出し、さらに、申請KECActive-reqを復号する。

【0072】次に、KEC230は、復号した申請KECActive-reqの内容を審理し、OKならば、CA220に、鍵回復有効要求321に対する応答322を返信する。当該応答322には、申請結果通知KECActive-resが含まれる。さらに、無効化された第1の公開鍵証明書の公開鍵と対の秘密鍵を示す鍵供託識別子KE-IdAが含まれる。このとき、鍵回復有効要求321に対する応答322をワンタイムに限るために、たとえば、CA22

10

20

30

40

50

0から送られてきた乱数 $r_3$ を基に、申請結果通知KECActive-resを暗号化することが好ましい。

【0073】また、KEC230は、鍵供託情報テーブル234に格納された、鍵供託識別子KE-Id<sub>A</sub>に対応付けられた第1の公開鍵証明書（あるいは第1の公開鍵証明書を示す識別子）の内容を、鍵回復有効状態に更新する。

【0074】CA220は、KEC230から鍵回復有効要求321に対する応答322を受信すると、USC200に、公開鍵証明書の再発行要求311に対する応答312を返信する。当該応答312には、申請結果通知CertRenewal-resが含まれる。さらに、再発行した公開鍵証明書（第2の公開鍵証明書）Certificateと、鍵供託識別子KE-Id<sub>A</sub>とが含まれる。このとき、第1の公開鍵証明書が廃止されたため、USC200には信頼できる公開鍵が存在しないので、ワンタイムな認証手続きの実現手段として、たとえば、USC200から送られてきた乱数 $r_2$ を基に申請結果通知CertRenewal-resを暗号化することが好ましい。

【0075】このように、図3に示す公開鍵更新手順では、USC200からの公開鍵証明書の再発行要求311を受け取ると、その延長として、KEC230に対して鍵回復要求321を通知している。この点が従来の公開鍵更新手順と異なる。

【0076】なお、図3では、第1の公開鍵証明書の無効化と第2の公開鍵証明書の発行（公開鍵証明書の再発行）との両方を確認した後に鍵回復を有効にするため、公開鍵証明書の再発行要求311に応答して、鍵回復有効要求321を起動するようにしている。しかしながら、公開鍵証明書の再発行要求311ではなく、第1の公開鍵証明書の無効化要求301に応答して、鍵回復有効要求321を起動するようにしてもよい。

【0077】以上、第1の公開鍵証明書を発行してから、当該証明書を廃止して第2の公開鍵証明書を再発行するまでの処理手順について説明した。

【0078】次に、利用者が、再発行を受けた第2の公開鍵証明書をを用いて認証を受けることで、廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵をカプセル開放してもらうまでの処理手順について、図4を参照して説明する。

【0079】図4は、本発明の第1実施形態において、利用者を認証してから、廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵をカプセル開放するまでの処理手順を説明するためのフロー図である。

【0080】まず、USC200は、利用者から廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵の開放要求を受け取ると、暗号データ鍵回復・再暗号化プログラム208を実行して、対象となるエンベロープデータ（廃止された第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵と、当該セッション

鍵で暗号化されたデータとを含む）、および、CA220から受け取った、廃止された第1の公開鍵証明書の公開鍵と対の秘密鍵を示す鍵供託識別子KE-Id<sub>A</sub>からなる鍵回復要求の申請を作成する。次に、電子署名プログラム202を実行して、CA220が再発行した公開鍵証明書（第2の公開鍵証明書）の公開鍵と対の秘密鍵を用いて、前記申請に署名してDRC240に送信する（ステップ140）。

【0081】DRC240は、鍵回復要求の申請を受け取ると、回復者認証プログラム242を実行し、当該申請書に付された公開鍵証明書（CA220が再発行した第2の公開鍵証明書）に記された公開鍵を用いて、当該申請書の署名を検証する（ステップ142）。その結果、署名が当該申請書に付された公開鍵証明書の公開鍵と対の秘密鍵によるものでないことが判明した場合は、エラー処理を行う（ステップ148）。

【0082】一方、署名が当該申請書に付された公開鍵証明書の公開鍵と対の秘密鍵によるものであることが判明した場合は、当該申請書の送信者が、鍵回復権限をもつ利用者であるか否かを必要に応じて検証する（ステップ144）。この鍵回復権限をもつ利用者であるか否かの検証は、たとえば、鍵供託識別子を用いて行う。本実施形態では、図2のステップ135において、CA220がUSC200に送付された鍵供託識別子KE-Id<sub>A</sub>を、ステップ140において、USC200からDRC240へ送信している。そこで、DRC240は、ステップ142において、署名が申請書に付された公開鍵証明書の公開鍵と対の秘密鍵によるものであることが判明した場合、当該申請書に付された鍵供託識別子KE-Id<sub>A</sub>をKEC230に送信する。これを受けて、KEC230は、鍵供託情報テーブル参照・登録・更新プログラム236を実行し、鍵供託情報テーブル234を参照して、受け取った鍵供託識別子KE-Id<sub>A</sub>に対応付けられた公開鍵証明書（第1の公開鍵証明書）が鍵回復をしてもよい（鍵回復有効）状態にあるか否かを判定する。そして、その判定結果を回復者認証プログラム242の戻り値に含めて、利用者認証を行う。

【0083】ステップ144において、回復権限をもつ利用者として認証された場合、すなわち、利用者が提供した鍵供託識別子KE-Id<sub>A</sub>を用いた鍵供託情報テーブル234の参照より、KEC230に鍵が保管され且つ鍵回復をしてもよい状態であると判定された場合には、ステップ150に示すセッション鍵回復処理が行われる（ステップ146）。一方、回復権限をもつ利用者として認証されなかった場合は、ステップ148に示すエラー処理が行われる（ステップ146）。

【0084】ステップ150のセッション鍵回復処理において、DRC240は、暗号データ鍵回復・暗号化プログラム243の実行を開始し、KEC230に対して、鍵供託識別子KE-Id<sub>A</sub>により特定される鍵（第1の

公開鍵証明書の公開鍵と対の秘密鍵)の検索を依頼する。これを受けて、KEC230は、秘密鍵検索プログラム237を実行して、鍵供託識別子KE-Id<sub>A</sub>により特定される鍵を検索する(ステップ152)。本実施形態では、供託鍵識別子がユニークであることを前提としているので、供託鍵識別子自体、あるいは当該供託鍵識別子から導出される識別子(保管鍵識別子)に基づいて検索することができる。

【0085】次に、DRC240は、ステップ152で検索された、第1の公開鍵証明書の公開鍵と対の秘密鍵を用いて、USC200から送られてきた鍵回復要求の申請書に付されたセッション鍵(第1の公開鍵証明書の公開鍵でカプセル化された鍵)を復号する(ステップ154)。

【0086】次に、DRC240は、鍵回復要求の申請書に付されたセッション鍵のカプセル開放のみならず、当該申請書に付された暗号データの復号をも行うか否か判断する(ステップ160)。この判断は、たとえば、鍵回復要求の申請書にカプセル化されたセッション鍵および当該セッション鍵で暗号化されたデータの両方が付されている場合は、暗号データの復号を行うものと判断し、カプセル化されたセッション鍵のみが付されている場合は、暗号データの復号を行わないものと判断するようにしてもよい。

【0087】鍵回復要求の申請書に付された暗号データの復号を行うと判断した場合は、ステップ170に示す暗号データ復号・再暗号化処理が行われる。一方、行わないと判断した場合は、フローを終了する。

【0088】ステップ170において、DRC240は、暗号データ鍵回復・暗号化プログラム243により、ステップ154で復号されたセッション鍵を用いて、鍵回復要求の申請書に付された暗号データを復号する(ステップ172)。次に、再暗号化用のセッション鍵を生成し(ステップ174)、ステップ172で復号化されたデータ(平文)を、前記再暗号化用のセッション鍵を用いて暗号化する(ステップ176)。さらに、USC200の利用者に対して、CA220が再発行した第2の公開鍵証明書の公開鍵を用いて、前記再暗号化用のセッション鍵をカプセル化し、これをステップ176で暗号化したデータに添付する。これにより、エンベロープデータを生成する(ステップ178)。そして、生成したエンベロープデータをUSC200に送付して、フローを終了する。

【0089】なお、図4に示すフローでは、エンベロープデータを対象にしたものについて説明したが、セッション鍵と暗号データとを別途に送る暗号方式のスキームでも、同様に処理できることはいうまでもない。

【0090】次に、上記説明した図4のフローのステップ144～ステップ170(鍵回復手順)におけるデータの流れについて説明する。

【0091】図5は、図4に示すフローのステップ144～ステップ170(鍵回復手順)におけるデータの流れを概念的に示した図である。

【0092】まず、USC200は、DRC240に対して、廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を用いた鍵回復が可能な状態にあるか否かの状態問い合わせである鍵回復可能確認要求要求401を送信する。当該要求401には、廃止した第1の公開鍵証明書の公開鍵と対の秘密鍵を用いた鍵回復が可能な状態にあるか否かの状態問い合わせ申請?CanKRS-reqと供託鍵識別子KE-Id<sub>A</sub>とを乱数 $r$ で暗号化したものと、乱数 $r$ をDRC240の公開鍵DRCpubでカプセル化したものとが含まれる。

【0093】DRC240は、鍵回復可能確認要求要求401を受信すると、DRCの秘密鍵DRCpriを用いて乱数 $r$ を取り出し、さらに申請?CanKRS-reqおよび供託鍵識別子KE-Id<sub>A</sub>を復号する。そして、KEC230に、供託鍵識別子KE-Id<sub>A</sub>が示す供託鍵(第1の公開鍵証明書の公開鍵と対の秘密鍵)を用いた鍵回復が、有効な状態にあるか否かの問い合わせである供託鍵ID有効確認要求421を送信する。当該要求421には、供託鍵識別子KE-Id<sub>A</sub>を用いた鍵回復が有効であるか否かを問い合わせるための申請KECStatus-reqと供託鍵識別子KE-Id<sub>A</sub>とを乱数 $r_2$ で暗号化したものと、乱数 $r_2$ をKEC230の公開鍵KECpubでカプセル化したものとが含まれる。

【0094】KEC230は、供託鍵ID有効確認要求421を受信すると、KECの秘密鍵KECpriを用いて乱数 $r_2$ を取り出し、さらに申請KECStatus-reqおよび供託鍵識別子KE-Id<sub>A</sub>を復号する。そして、鍵供託情報テーブル234を参照して、復号した鍵供託識別子KE-Id<sub>A</sub>に対応付けられた公開鍵証明書(第1の公開鍵証明書)が鍵回復をしてもよい(鍵回復有効)状態にあるか否かを判定する。そして、DRC240に、供託鍵ID有効確認要求421に対する応答422を返信する。この応答には、鍵供託識別子KE-Id<sub>A</sub>が示す秘密鍵(第1の公開鍵証明書の公開鍵と対の秘密鍵)による鍵回復についての有効/無効の判定結果通知KECStatus-resが含まれる。このとき、ワンタイムな認証手続きの実現手段として、たとえば、DRCから送られてきた乱数 $r_2$ を基に、判定結果通知KECStatus-resを暗号化することが好ましい。

【0095】DRC240は、KEC230から応答422を受信すると、USC200に、鍵回復可能確認要求要求401に対する応答402を返信する。当該応答402には、KEC230からの応答422に含まれる判定結果通知KECStatus-resに対応した申請結果通知?CanKRS-resと、鍵供託識別子KE-Id<sub>A</sub>と、乱数 $r_3$ とが含まれる。このとき、ワンタイムな認証手続きの実現手段として、たとえば、USC200から送られてきた乱数 $r$ を基に、申請結果通知?CanKRS-res、鍵供託識別子KE-I

$d_A$ 、および乱数  $r_3$  を暗号化することが好ましい。あるいは、USC200の利用者に対して発行された第2の公開鍵証明書の公開鍵  $User_{Anewpub}$  を用いて暗号してもよい。

【0096】次に、USC200は、DRC240からの応答402に含まれる申請結果通知  $CanKRS-res$  が、鍵供託識別子  $KE-Id_A$  で特定される鍵を用いた鍵回復が有効であることを示している場合、DRC240に、鍵供託識別子  $KE-Id_A$  で特定される鍵を用いた鍵回復要求411を送信する。当該要求411には、暗号データ  $Ciphertext$  (平文をセッション鍵SKで暗号化したものと、セッション鍵SKを供託識別子  $KE-Id_A$  が示す鍵 (第1の公開鍵証明書の公開鍵と対の秘密鍵  $User_{Aoldpri}$ ) でカプセル化したものを含む) と、鍵回復要求を示す申請  $KeyRecovery-req$  および供託鍵識別子  $KE-Id_A$  を、応答402に含まれる乱数  $r_3$  で暗号化したものと、乱数  $r_3$  をDRC240の公開鍵  $DRCpub$  でカプセル化したものが含まれる。

【0097】DRC240は、USC200から鍵回復要求411を受信すると、DRC240の秘密鍵  $DRCpri$  を用いて乱数  $r_3$  を取り出し、さらに申請  $KeyRecovery-req$  および供託鍵識別子  $KE-Id_A$  を復号する。ここで、上記説明した鍵回復可能確認要求要求401と鍵回復要求411とが、同じセッションで行われるのであれば、乱数  $r_3$  は、鍵回復可能確認要求要求401で用いた乱数  $r$  と同じものでもよい。

【0098】次に、DRC240は、KEC230から受け取った応答422に含まれる、供託鍵識別子  $KE-Id_A$  により特定される鍵を用いた鍵復号についての有効/無効の判定結果通知  $KECstatus-res$  を調べ、該通知の内容が有効であれば、KEC230に、セッション鍵回復要求431を送信する。当該要求431には、セッション鍵SKを第1の公開鍵証明書の公開鍵  $User_{AoldPub}$  でカプセル化したもの、復号化することを許可することを示す申請  $KECRecovery-req$ 、および、供託鍵識別子  $KE-Id_A$  を乱数  $r_4$  で暗号化したものと、乱数  $r_4$  をKEC230の公開鍵  $KECpub$  でカプセル化したものが含まれる。

【0099】KEC230は、DRC240からセッション鍵回復要求431を受信すると、KEC230の秘密鍵  $KECpri$  を用いて乱数  $r_4$  を取り出し、さらに申請  $KECRecovery-req$ 、供託鍵識別子  $KE-Id_A$ 、および公開鍵  $User_{AoldPub}$  でカプセル化したセッション鍵SKを復号する。次に、供託鍵識別子  $KE-Id_A$  により特定される鍵 (第1の公開鍵証明書の公開鍵と対の秘密鍵  $User_{Aoldpri}$ ) を検索し、この鍵を用いて、セッション鍵SKをカプセル開放する。

【0100】それから、KEC230は、DRC240に、セッション鍵回復要求431に対する応答432を返信する。当該応答432には、鍵回復が行われたか否かを示す申請結果通知  $KECRecovery-res$  と、カプセル開

放されたセッション鍵SKとが含まれる。このとき、セッション鍵回復要求431に対する応答432をワントタイムに限るために、たとえば、DRC240から送られてきた乱数  $r_4$  を基に、申請結果通知  $KECActive-res$ 、およびセッション鍵SKを暗号化することが好ましい。

【0101】DRC240は、KEC230から応答432を受信すると、当該応答432に含まれるセッション鍵SKを用いて、USC200からの鍵回復要求411に含まれる暗号データ  $Ciphertext$  を平文  $Plaintext$  に復号する。それから、USC200に、鍵回復要求411に対する応答412を返信する。当該応答412には、KEC230から受け取った応答432に含まれる申請結果通知  $KeyRecovery-res$  と、平文  $Plaintext$  を乱数  $r_5$  で暗号化したものと、乱数  $r_5$  をUSC200の利用者に再発行された第2の公開鍵証明書の公開鍵  $User_{AnewPub}$  でカプセル化したものが含まれる。このとき、ワントタイムな認証手続きの実現手段として、たとえば、USC200から送られてきた乱数  $r_3$  を基に、申請結果通知  $KeyRecovery-res$  を暗号化することが好ましい。

【0102】このように、図5に示す鍵回復手順では、DRC240は、USC200からの鍵回復可能確認要求401を受け取ると、当該要求401に付された供託鍵識別子を用いて、当該識別子により特定される鍵 (第1の公開鍵証明書の公開鍵と対の秘密鍵) を用いた鍵回復が有効であるか否かの判定を行うようにしている。そして、有効である場合には、当該鍵を用いてセッション鍵をカプセル開放し、当該セッション鍵で暗号化されたデータを復号化するとともに、当該復号化されたデータを新たなセッション鍵で再暗号化し、さらに、第1の公開鍵証明書を廃止した際に発行された第2の公開鍵証明書の公開鍵を用いて、当該新たなセッション鍵を再カプセル化している。この点が従来の鍵回復手順と異なる。

【0103】以上、利用者が、再発行を受けた第2の公開鍵証明書の公開鍵を用いて認証を受けることで、廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵をカプセル開放してもらうまでの処理手順について説明した。

【0104】上記の実施形態では、CA220は、第1の公開鍵証明書 (データ暗復号用) を発行した利用者が当該証明書の公開鍵と対の秘密鍵を紛失した場合、当該利用者から要求があった場合にのみ、当該証明書を廃止して、前記利用者に第2の公開鍵証明書 (データ暗復号兼認証用) を発行するようにしている。

【0105】そして、DRC240は、前記第2の公開鍵証明書の公開鍵を用いて利用者を認証することにより、前記第2の公開鍵証明書を発行を受けた利用者により、当該利用者が第1の公開鍵証明書の発行の際にKEC230に供託した公開鍵と対の秘密鍵を用いた鍵回復を認めるようにしている。

【0106】このように、本実施形態では、鍵回復権限

を有する利用者を当該利用者に発行した第2の公開鍵証明書を用いて認証するので、鍵回復の利用者を登録する必要がない。すなわち、公開鍵証明書の利用者の認証・登録と鍵回復の利用者の認証・登録とを一元管理することができる。また、CA220による公開鍵証明書の発行を受けた利用者に対してのみ、当該公開鍵証明書の公開鍵と対の秘密鍵を紛失した場合に、当該秘密鍵を用いた鍵回復を認めることができる。

【0107】また、本実施形態では、DRC240は、鍵回復権限を有する利用者の要求により、回復したセッション鍵を用いて暗号データを復号した後、この復号したデータを新たなセッション鍵で再暗号化し、さらに、この新たなセッション鍵を、CA220が当該利用者に再発行した第2の公開鍵証明書の公開鍵を用いてカプセル化して、当該利用者のUSC200に送信するようにしている。

【0108】このようにすることで、回復したデータを利用者に通知する際のセキュリティを向上させることができる。

【0109】以上、本発明の第1実施形態について説明した。

【0110】次に、本発明の第2実施形態について説明する。

【0111】上記の第1実施形態では、利用者を当該利用者に発行した第2の公開鍵証明書を用いて認証し、認証された場合に鍵回復を認めている。これに対し、本実施形態では、鍵回復の有効期限を設定しておき、認証された場合でも、有効期限が切れている場合は鍵回復を認めないようにしている。

【0112】本実施形態は、上記の第1実施形態に対し、図2に示すステップ135での処理と、図4に示すステップ152での処理が異なる。その他については、上記の第1実施形態と同じであるので、図2に示すステップ135および図4に示すステップ152に対応する処理についてのみ説明し、その他の説明は省略する。

【0113】図6は、本発明の第2実施形態における、図2のステップ135に対応するステップ135aでの処理を説明するためのフロー図である。

【0114】まず、CA220は、USC200から公開鍵証明書の再発行申請を受け取ると、公開鍵証明書申請・更新プログラム226の実行を開始し、管理者が、鍵回復の期間について、あらかじめ定義したシステム設定を読み込む(ステップ500)。このシステム設定とは、ファイルの形で与えられ、有効期限データ、有効期限データの送付方法(公開鍵証明書の拡張をする／しない、その他の方法)、鍵回復を認めない利用者のリスト等を記述したものである。このシステム設定ファイルは、計算機装置221の記憶装置に、あらかじめ格納されているものとする。

【0115】次に、CA220は、ステップ500で読

み込んだシステム設定にしたがい、有効期限を公開鍵証明書に添付するか否かを判定する(ステップ502)。添付しない場合は、別途送付を選択し(ステップ510)。有効期限を記憶媒体に記憶してオフラインでUSC200の利用者の送付するか、あるいは、公開鍵証明書の送信とは別に、オンラインでUSC200に送信する。この際、CA220の秘密鍵を用いて署名をしておくことが好ましい。

【0116】一方、公開鍵証明書に添付する場合は、さらに、公開鍵証明書の形式を拡張するか否かの判定を行なう(ステップ504)。

【0117】公開鍵証明書の形式を拡張する場合は、公開鍵証明書を作成するとともに、当該公開鍵証明書に、公開鍵証明書の拡張形式として鍵回復の有効期限に関する情報を追加する(ステップ506)。たとえば、公開鍵証明書x.509Version3の拡張形式extensionsとして、供託鍵識別子KeyEscrowIdentifierおよび鍵回復の有効期限KeyRecoveryUsagePeriodの情報を添付する。

【0118】公開鍵証明書の形式を拡張しない場合は、公開鍵証明書を作成するとともに、当該公開鍵証明書の有効期限Validityを鍵回復の有効期限KeyRecoveryUsagePeriodの代わりとする(ステップ507)。

【0119】次に、CA220は、作成した公開鍵証明書を第2の公開鍵証明書として再発行して、これをUSC200に返信する。また、発行した第2の公開鍵証明書をデータベース222に格納する(ステップ508)。

また、CA220は、第2の公開鍵証明書と結合する方法および別途送付する方法のいずれかにより、第1の公開鍵証明書の公開鍵と対の秘密鍵がKEC230に供託された証拠を示す付加情報および当該秘密鍵を示す鍵供託識別子を、USC200に送付する。

【0120】上記のフローにより、公開鍵証明書の再発行に際し、鍵回復の有効期限が設定されることになる。

【0121】なお、上記のステップ507において、公開鍵証明書の有効期限を、そのまま鍵回復の有効期限とするのではなく、評価式、たとえば、公開鍵証明書に記された有効期限の1/12の期間が経過するまで(公開鍵証明書の有効期限が1年であれば最初の30日が経過するまで)を鍵回復の有効期限とする評価式を、CA220とDRC240との間で取り交わしておくようにしてもよい。

【0122】また、上記のステップ510において、CA220は、自らの署名が付いた鍵回復の有効期限の情報を、USC200に送付している。したがって、USC200の利用者は、CA220の署名が付いた有効期限の情報に、さらに自らの署名を付けて、DRC240に通知することになる。しかしながら、ステップ510において、CA220の署名が付いた鍵回復の有効期限の情報を、CA220からDRC240に直接送付するようにしてもよい。

【0123】図7は、本発明の第2実施形態における、図4のステップ152に対応するステップ152aでの処理を説明するためのフロー図である。

【0124】まず、DRC240は、暗号データ鍵回復・再暗号化プログラム243の実行を開始し、管理者が、鍵回復の期間について、あらかじめ定義したシステム設定を読み込む（ステップ550）。このシステム設定とは、ステップ500でのシステム設定と同じものであり、ファイル形式で、計算機装置241の記憶装置に、あらかじめ格納されているものとする。

【0125】次に、DRC240は、ステップ550で読み込んだシステム設定にしたがい、鍵回復の有効期限が公開鍵証明書に添付されている否かを判定する（ステップ552）。添付されていない場合は、別途送付を選択し（ステップ560）。USC200の利用者から有効期限を記憶した記憶媒体がオフラインで送付されてくるか、あるいは、公開鍵証明書とは別に、オンラインで送信されてくるのを待つ。この際、USC220の利用者は、たとえば再発行された第2の公開鍵証明書の公開鍵と対の秘密鍵を用いて署名をしておくことが好ましい。

【0126】一方、鍵回復の有効期限が公開鍵証明書に添付されている場合は、さらに、公開鍵証明書の形式が拡張されているか否かの判定を行なう（ステップ554）。

【0127】公開鍵証明書の形式が拡張されている場合は、第2の公開鍵証明書に拡張形式として追加された鍵回復の有効期限に関する情報を読み込む（ステップ556）。たとえば、公開鍵証明書x.509Version3の拡張形式extensionsとして添付された、供託鍵識別子KeyEscrowIdentifierおよび鍵回復の有効期限KeyRecoveryUsagePeriodの情報を読み込む。

【0128】公開鍵証明書の形式が拡張されていない場合は、第2の公開鍵証明書の有効期限Validityを鍵回復の有効期限KeyRecoveryUsagePeriod、あるいは鍵回復の有効期限を特定するためのパラメータとして読み込む（ステップ557）。

【0129】次に、DRC240は、上記のステップにより得られた鍵回復の有効期限を経過しているか否かを判定し、経過していない場合にのみ、KEC230に対して、鍵供託識別子KE-IdAにより特定される鍵（第1の公開鍵証明書の公開鍵と対の秘密鍵）の検索を依頼する。これを受けて、KEC230は、秘密鍵検索プログラム237を実行して、鍵供託識別子KE-IdAにより特定される鍵を検索する（ステップ152）。上述したように、供託鍵識別子はユニークであることを前提としているので、供託鍵識別子自体、あるいは当該供託鍵識別子から導出される識別子（保管鍵識別子）に基づいて検索することができる。

【0130】上記のフローにより、鍵回復の要求に際

し、鍵回復の有効期限内にある場合にのみ、鍵回復が行われることになる。

【0131】以上、本発明の第2実施形態について説明した。

【0132】次に、本発明の第3実施形態について説明する。

【0133】上記の第2実施形態では、鍵回復の有効期限の情報をCA220が再発行した第2の公開鍵証明書に添付、あるいは、鍵回復の有効期限自体を別途送付することで、鍵回復の有効期限を設定するようにしている。これに対し、本実施形態では、供託鍵識別子を用いて有効期限を設定するようにしている。

【0134】本実施形態は、上記の第1実施形態に対し、図4に示すステップ150での処理が異なる。その他については、上記の第1実施形態と同じであるので、図4のステップ150に対応する処理についてのみ説明し、その他の説明は省略する。

【0135】図8は、本発明の第3実施形態における、図4のステップ150に対応するステップ150aでの処理を説明するためのフロー図である。

【0136】図8に示すステップ150aが図4のステップ150と異なる点は、ステップ152の前処理としてステップ600を設けた点である。そこで、以下では、ステップ600での処理についてのみ説明する。

【0137】まず、DRC240は、暗号データ鍵回復・暗号化プログラム243の実行を開始し、鍵供託識別子KE-IdAについて、鍵回復の有効期限が無制限に設定されているか否かを判定する（ステップ601）。判定方法としては、図2に示すステップ135での処理において、所定の基準にしたがい鍵回復の有効期限を無制限／制限ありに設定して、この情報を供託鍵識別子に付加してUSC200に送付しておき、USC200から受け取った供託鍵識別子から当該鍵回復の有効期限情報を取り出して判断するか、あるいは、計算機装置221が備える記憶装置に、USC200に送付した供託鍵識別子と、所定の基準にしたがい無制限／制限ありに設定された鍵回復の有効期限情報とを対応付けて記憶しておき、USC200から受け取った供託鍵識別子に対応する鍵回復の有効期限情報を当該記憶装置から取り出して判断する方法などが考えられる。

【0138】ステップ601において、鍵回復の有効期限情報が無制限である場合には、ステップ152に移行する。一方、無制限でない場合は、現在の時刻（システムの運用規則によっては、利用者が鍵回復を要求した時刻でもよい）Tcurrentと、供託停止後、継続して供託鍵を保管する義務がある期間として予め定められた期間Tlimitとを取得する（ステップ612）。次に、供託鍵識別子の供託開始時刻ID-t1および停止時刻ID-t2を取得する（ステップ614）。ここで、供託鍵識別子の供託開始時刻ID-t1とは、CA220が第1の公開鍵証明書

を発行した時刻、あるいは、KEC230が第1の公開鍵証明書の公開鍵と対の秘密鍵についての供託識別子を発行した時刻を意味し、供託停止時刻ID-t2とは、CA220が第1の公開鍵証明書を廃止した時刻、あるいは、KEC230が、CA220による第2の公開鍵証明書の再発行に際して、1の公開鍵証明書の公開鍵と対の秘密鍵を用いた鍵回復が有効となるように、供託鍵情報テーブル234を更新した時刻を意味する。

【0139】次に、DRC240は、鍵回復の対象となる暗号データが暗号化された時刻tを取得する(ステップ616)。たとえば、暗号データがファイル形式で格納されているのであれば、そのファイルの作成日時を入力する。

【0140】次に、DRC240は、ステップ616で取得した時刻tが、供託開始時刻ID-t1から供託停止時刻ID-t2に供託停止後の保管義務がある期間Tlimitを加算した期間までの範囲に含まれているか否かを判定する(ステップ618)。

【0141】含まれない場合は、鍵回復不可としてエラー処理を行う(ステップ630)。一方、含まれている場合は、ステップ612で取得した時刻Tcurrentが、供託停止時刻ID-t2から供託停止時刻ID-t2に供託停止後の保管義務がある期間Tlimitを加算した期間までの範囲に含まれているか否かを判定する(ステップ619)。含まれていない場合は、鍵回復不可としてエラー処理を行う(ステップ630)。一方、含まれている場合は、鍵回復を許可するものとし(ステップ620)、ステップ152に移行する。

【0142】上記のフローにより、鍵回復の要求に際し、鍵回復の有効期限内にある場合にのみ、鍵回復が行われることになる。

【0143】なお、上記のフローでは、鍵回復の有効期限を一律に設定しているが、当該期限は個々のシステム運用に応じて変形するようにしてもよいことはいふまでもない。たとえば、以下のように変形してもよい。

【0144】(a) オンラインでの回復は一律に設定される有効期限内に限定とし、オフラインでの回復は、専門の職員による面接で判断する。

【0145】(b) 特定の権限(たとえば、公開鍵証明書に任意のクラス以上の権限を有することが示されている場合)をもつ利用者に対しては、鍵回復の有効期限(すなわち、供託停止後の保管義務がある期間Tlimit)を長く設定しておく。

【0146】(c) 特定の権限をもたない利用者に対しては、オフラインでの回復を原則とし、本処理の適応範囲外とする。

【0147】(d) 暗号データの種類に応じて鍵回復の有効期限(すなわち、供託停止後の保管義務がある期間Tlimit)の設定を変える。

【0148】本実施形態によれば、秘密鍵の供託開始／

停止は、利用者の責任で処理することができる。また、供託停止後、供託鍵を保管する期間の設定は、DRC240の管理者の責任で定めることができる。また、利用者がKEC230に供託した鍵を紛失した場合にのみ、当該利用者に対して、当該鍵を用いた鍵回復を提供することができる。

【0149】以上、本発明の第3実施形態について説明した。

【0150】次に、本発明の第4実施形態について説明する。

【0151】公開鍵証明書には、通常、有効期限が設定されており、当該期限が過ぎると、当該公開鍵証明書は廃止されて、第2世代の公開鍵証明書が発行される。すなわち、上記の実施形態でいえば、データ暗復号用の第1の公開鍵証明書が多世代にわたり存在することが考えられる。

【0152】このような状況下で、利用者が多世代にわたり複数発行された第1の公開鍵証明書の公開鍵と対の秘密鍵各々をKEC230に供託する場合、供託する秘密鍵各々に対して別個の供託鍵識別子を付して管理するよりも、同じ供託鍵識別子を用いて管理することが好ましい。

【0153】供託鍵識別子は、鍵回復の権限を有することを保証するための機密データであり、ICカードなどに格納して保管することが望まれる。しかしながら、ICカードなどの媒体の記憶容量では、供託鍵識別子および公開鍵証明書(たとえば、512BYTE)を複数格納することが困難な場合(たとえば1K BYTE)が多い。このため、最初に発行された第1の公開鍵証明書の公開鍵と対の秘密鍵を供託した際に発行された供託鍵識別子をもとに、2回目以降に発行された第1の公開鍵証明書の公開鍵と対の秘密鍵の供託を管理することが望まれる。

【0154】また、利用者にとっては、相手から、多世代にわたり複数発行された第1の公開鍵証明書のうちの1つの公開鍵を用いて暗号化されたデータを受け取った場合、対応する秘密鍵が分からなくなってしまうデータを復号できなくなってしまう場合も考えられる。

【0155】本実施形態では、このような場合を想定したものであり、多世代にわたり複数発行された第1の公開鍵証明書の公開鍵と対の秘密鍵各々について、別個の供託鍵識別子を付すことなく、これらの秘密鍵の供託を管理することができるようにしている。また、利用者が多世代にわたり複数発行された第1の公開鍵証明書の公開鍵と対の秘密鍵をKEC230に供託している場合に、当該利用者が鍵回復を行うための秘密鍵が分からない場合でも鍵回復できるようにしている。

【0156】本実施形態は、上記の第1実施形態に対し、図2に示すステップ110での処理において、多世代にわたり複数発行された第1の公開鍵証明書(データ暗復号用)の公開鍵と対の秘密鍵各々を供託する場合



に、これら秘密鍵に同じ供託鍵識別子を付加する点、および、図4に示すステップ152での処理が異なる。その他については、上記の第1実施形態と同じであるので、図4のステップ152に対応する処理についてののみ説明し、その他の説明は省略する。

【0157】図9は、本発明の第4実施形態における、図4のステップ152に対応するステップ152bでの処理を説明するためのフロー図である。

【0158】まず、KEC230は、秘密鍵検索プログラム237を実行して、USC200から受け取った供託鍵識別子に複数の版があるか否かを判定する(ステップ710)。判定方法としては、図2に示すステップ110での処理において、多世代にわたり複数発行された第1の公開鍵証明書の公開鍵と対の秘密鍵各々に同じ供託鍵識別子を付加する際に、当該供託鍵識別子に版(世代)情報を含めておき、USC200から受け取った供託鍵識別子に版情報が含まれるか否かで判断する方法や、USC200から受け取った供託鍵識別子と同じもの(ただし、版情報が違う)がKEC230の供託鍵情報テーブル234に複数格納されているか否かで判断する方法などが考えられる。

【0159】ステップ710において、版情報がない(たとえば、版番号が0に設定されている)場合は、ステップ700へ移行して、図4のステップ152と同じ処理を行い、USC200から受け取った供託鍵識別子により特定される秘密鍵を検索する。具体的には、USC200から受け取った供託鍵識別子を入力変数とする関数 $f()$ を計算し、秘密鍵の保管位置(たとえば、計算機装置232が備えるデータベースのレコード番号)を決定し、決定された保管位置から秘密鍵を取得する(ステップ700)。ここで、関数 $f()$ として、たとえば、一方向性関数を用いられる。なお、ハッシュ表探索として、保管位置が登録・参照できることが好ましい。

【0160】一方、ステップ710において、版情報がある(たとえば、版番号が1以上に設定されている)場合は、ステップ712に移行して、鍵回復の対象となる暗号データが暗号化された時刻 $t$ を取得する。たとえば、暗号データがファイル形式で格納されているのであれば、そのファイルの作成日時を入力する。

【0161】次に、USC200から受け取った供託鍵識別子について、1から $n$ 番目までの版うち、どの版の秘密鍵を検索すべきかシーケンシャルに解析する。

【0162】まず、KEC230は、鍵供託情報テーブル参照・登録・更新プログラム236を実行して鍵供託情報テーブル234を参照し、USC200から受け取った供託鍵識別子に対して、各版番号 $i=1\sim n$ 毎に、鍵回復が許されるかどうかを判定する(ステップ714)。鍵回復が有効であれば、その版番号 $i$ が付された供託鍵識別子の登録時刻 $T_i$ と、つぎの版番号 $i=i+1$ が付された供託鍵識別子の登録時刻 $T_{i+1}$ とを取得する

(ステップ716)。一方、無効であれば、つぎの版番号 $i=i+1$ についての情報を取得するため、ステップ722に移行する。

【0163】ステップ716での処理終了後、KEC230は、ステップ716で取得した、版番号 $i$ が付された供託鍵識別子の登録時刻 $T_i$ と、版番号 $i+1$ が付された供託鍵識別子の登録時刻 $T_{i+1}$ との間に、ステップ712で取得した時刻 $t$ が含まれるか否かを判断する(ステップ718)。含まれていれば、版番号 $i$ の供託鍵情報識別子により特定される秘密鍵を用いた鍵回復が可能であると判定して、ステップ720に移行する。

【0164】一方、含まれていない場合は、版番号 $i$ の供託鍵情報識別子により特定される秘密鍵を用いた鍵回復が不可能であると判定し、つぎの版番号 $i=i+1$ についての情報を取得するため、ステップ722に移行する。

【0165】ステップ720では、KEC230は、版番号 $i$ の供託鍵識別子と当該識別子の登録時刻 $T_i$ とを入力変数として、ステップ700と同じ関数 $f()$ を計算し、秘密鍵の保管位置(たとえば、計算機装置232が備えるデータベースのレコード番号)を決定し、決定された保管位置から秘密鍵を取得する。

【0166】ステップ722では、次の版番号 $i+1$ が最大数 $n$ 以下であるか否かを調べ、 $n$ 以下の場合には、 $i$ を1つインクリメントし(ステップ724)、ステップ714に戻る。一方、次の版番号 $i+1$ が最大数 $n$ を超えた場合は、エラー処理として終了する(ステップ730)。

【0167】以上、本発明の第4実施形態について説明した。

【0168】なお、上記の各実施形態では、利用者が紛失した秘密鍵を用いて、当該秘密鍵と対の公開鍵でカプセル化されたセッション鍵を開放する場合について説明したが、本発明はこれに限定されるものではなく、利用者が紛失した秘密鍵を用いて、当該秘密鍵と対の公開鍵で暗号化されたデータを復号する場合すべてに適用可能なというまでもない。

【0169】最後、以上説明した本発明の各実施形態を、電子商取引システムに適用した場合について説明する。

【0170】図10は、本発明の各実施形態が適用された電子商取引システムの概略構成図である。

【0171】このシステムは、図示するように、送信装置801と、受信装置802と、暗号データ保存装置813と、会計監査機関861と、鍵回復機関851と、送信装置801および受信装置802間で通信を行うための通信路800と、で構成される。

【0172】鍵回復機関851は、図1のCA220、KEC230およびDRC240に相当するものである。受信装置802は、図1のUSC200に相当する

10

20

30

40

50

ものであり、USC200が搭載する各種プログラム202、204、206、208に加えて、暗号データを保存装置803に保存するための保存プログラムが搭載される。

【0173】送信装置801は、取引データを、受信装置201の利用者が公開鍵（第1の公開鍵証明書の公開鍵）を用いて暗号化して、受信装置201に送信するためのプログラムが搭載されている。会計監査機関860は、電子商取引に関して監査手順を実行する。

【0174】通常、電子商取引において、暗号化された取引データは、受信後、直ちに復号される。ただし、図10に示す例では、受信装置801は、送信装置802から送られてきた取引データを受け取ると、保存プログラムを実行して、たとえば、受信装置801の処理能力や作業進行に応じて、受け取った取引データを日/月単位で保存装置803に保管するようにしている。そして、後から、受信装置201の利用者の秘密鍵（第1の公開鍵証明書の秘密鍵）を用いて復号化することができるようにしている。

【0175】受信装置201の利用者が秘密鍵を紛失した場合は、受信装置201および鍵回復機関851間において、上記説明した各実施形態を実施することで、取引データの復号を行うことが可能となる。

【0176】なお、会計監査機関860は、保存装置803に暗号化された状態の取引データを用いて、監査手順を実施することも可能である。この場合、暗号化された状態の取引データを復号するための鍵の入手は、鍵回復機関851に依頼する。

【0177】

【発明の効果】以上説明したように、本発明によれば、公開鍵証明書および鍵回復における利用者管理を一元化し、公開鍵証明書の発行を受けた利用者に対してのみ、当該公開鍵証明書の公開鍵と対の秘密鍵を紛失した場合に、当該秘密鍵を用いたデータ回復を認めることができる。

【図面の簡単な説明】

【図1】本発明の第1実施形態が適用されたシステムの概略構成図である。

【図2】本発明の第1実施形態において、第1の公開鍵証明書を発行してから、当該証明書を廃止して第2の公開鍵証明書を再発行するまでの処理手順を説明するためのフロー図である。

【図3】図2のステップ130（公開鍵証明書の更新手順）におけるデータの流れを概念的に示した図である。

【図4】本発明の第1実施形態において、利用者を認証

してから、廃止した第1の公開鍵証明書の公開鍵でカプセル化されたセッション鍵をカプセル開放するまでの処理手順を説明するためのフロー図である。

【図5】図4に示すフローのステップ144～ステップ170（鍵回復手順）におけるデータの流れを概念的に示した図である。

【図6】本発明の第2実施形態における、図2のステップ135に対応するステップ135aでの処理を説明するためのフロー図である。

【図7】本発明の第2実施形態における、図4のステップ152に対応するステップ152aでの処理を説明するためのフロー図である。

【図8】本発明の第3実施形態における、図4のステップ150に対応するステップ150aでの処理を説明するためのフロー図である。

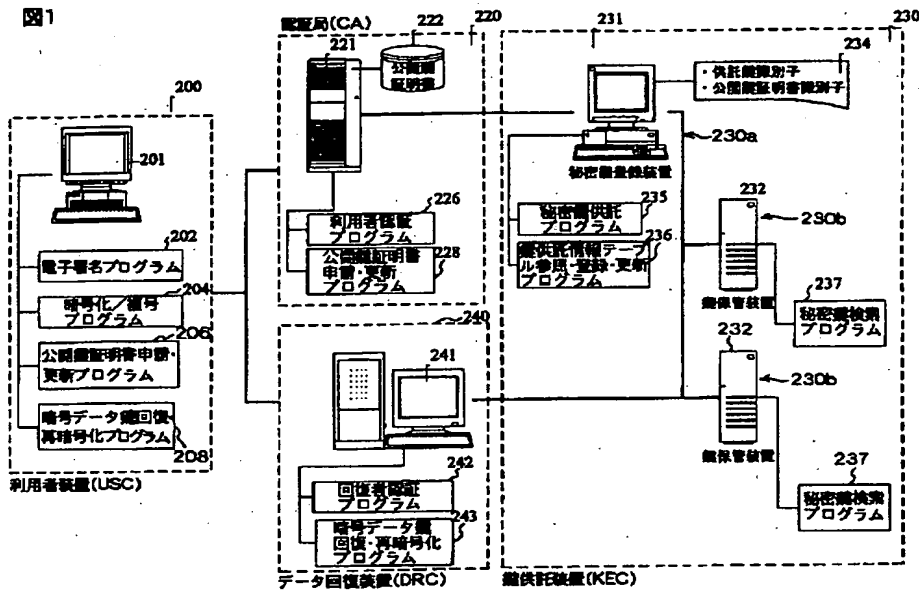
【図9】本発明の第4実施形態における、図4のステップ152に対応するステップ152bでの処理を説明するためのフロー図である。

【図10】本発明の各実施形態が適用された電子商取引システムの概略構成図である。

【符号の説明】

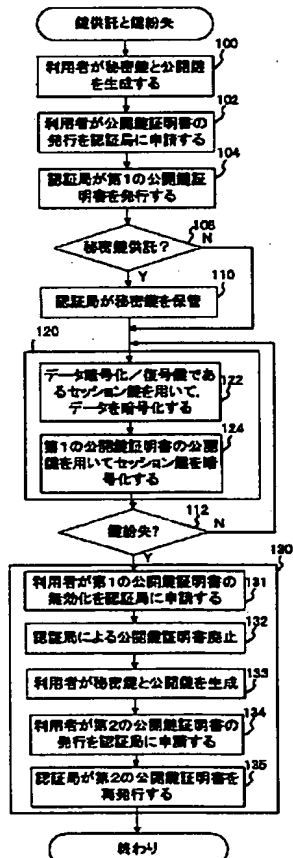
200 USC  
201、221、231、232、241 計算機装置  
202 電子署名プログラム  
204 暗号化/復号プログラム  
206、228 公開鍵証明書申請・更新プログラム  
208、243 暗号データ鍵回復・再暗号化プログラム  
222 データベース  
226 利用者認証プログラム  
230 KEC  
230a 秘密鍵登録装置  
230b 鍵保管装置  
234 鍵供託情報テーブル  
235 秘密鍵供託プログラム  
236 鍵供託情報テーブル参照・登録・更新プログラム  
237 秘密鍵検索プログラム  
242 回復者認証プログラム  
40 800 通信路  
801 送信装置  
802 受信装置  
803 保存装置  
851 鍵回復期間  
861 会計監査機関

【図1】



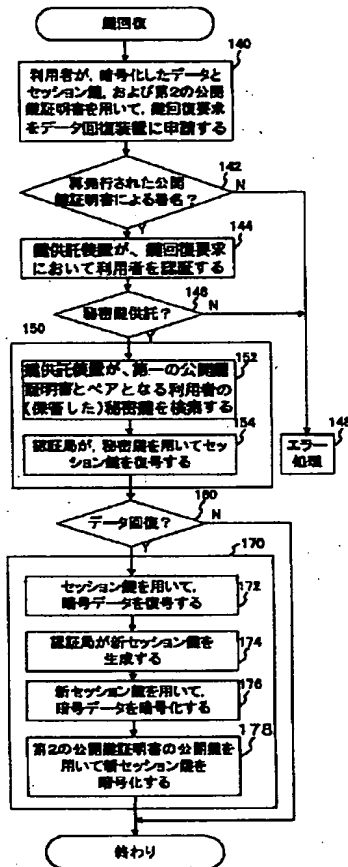
【図2】

図2

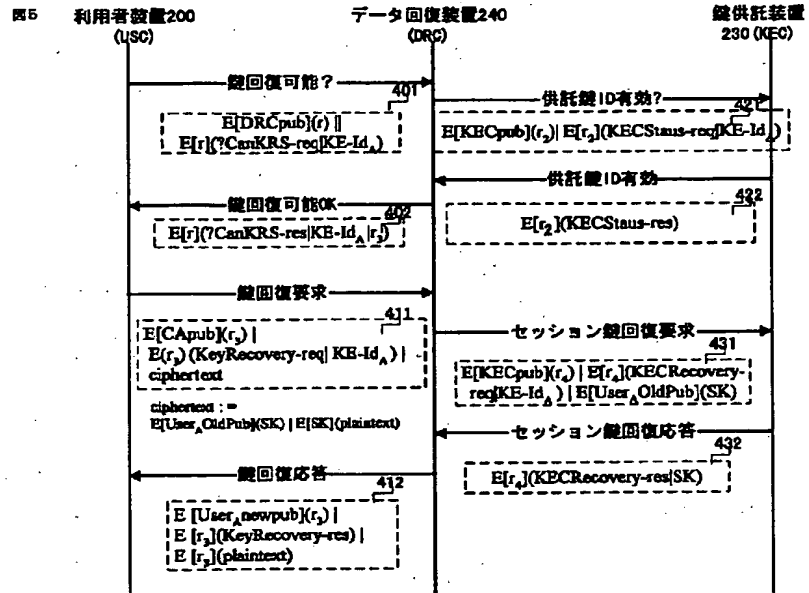


【図4】

図4

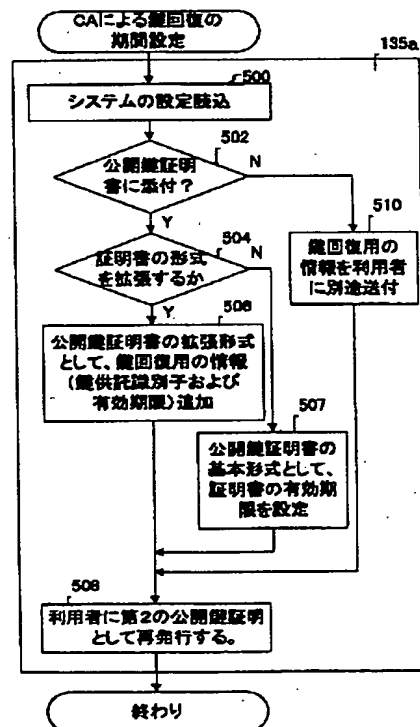


【図5】



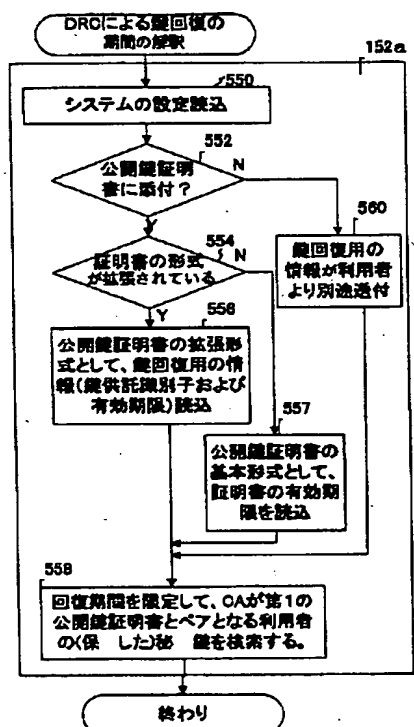
【図6】

図6



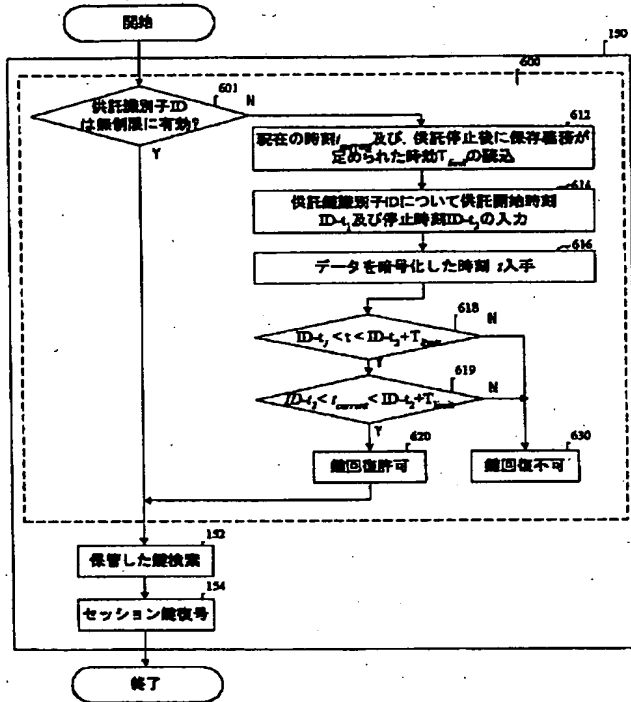
【図7】

図7



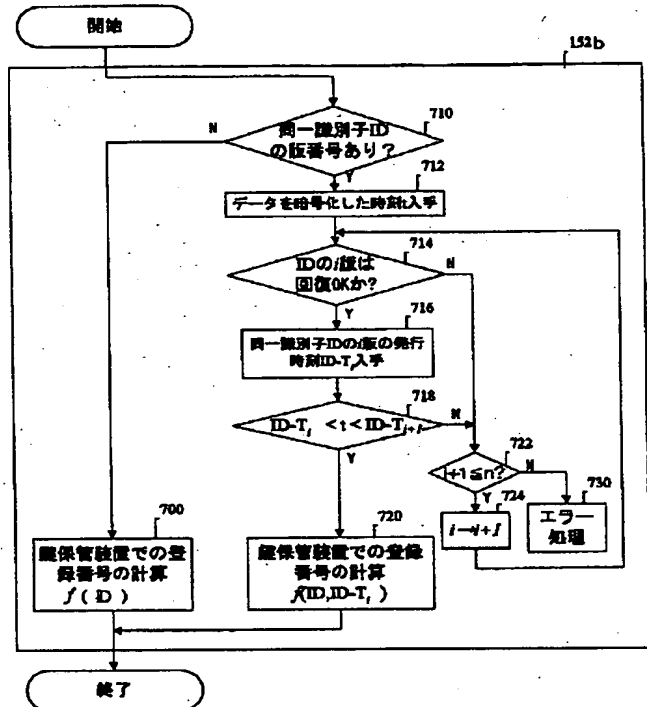
【図8】

図8



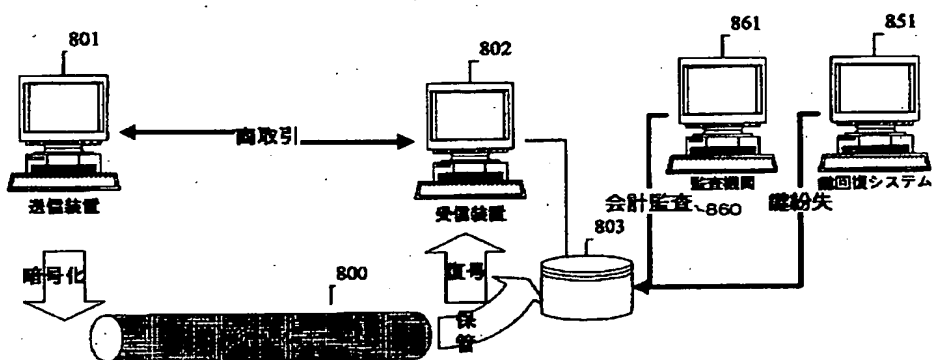
【図9】

図9



【図10】

図10



フロントページの続き

(72)発明者 川井 亨  
 神奈川県横浜市戸塚区品濃町504番地2  
 日立電子サービス株式会社内

(72)発明者 柳内 秀敬  
 神奈川県横浜市戸塚区品濃町504番地2  
 日立電子サービス株式会社内